

## Úvod

Dnešní život si již nedovedeme představit bez počítačů. Výpočetní technika a informační technologie vůbec dnes ovlivňují prakticky veškerou lidskou činnost. Jedním z prvních oborů, do kterého výpočetní technika začala významně zasahovat, bylo strojírenství. Již před půl stoletím se začaly objevovat první náznaky toho, co dnes označujeme všeobecně známým pojmem CAD (Computer Aided Design). Dnes pod tuto zkratku zahrnujeme nejen softwarové konstrukce modelů strojních součástí, ale i spoustu dalších užitečných prostředků, bez kterých si dnes již nedovedeme představit konstruktérskou praxi.

Vývoj algoritmů pro tyto systémy si od jejich vývojářů postupně vyžádal řadu specifických znalostí, které měly geometrickou povahu, které se však od poznatků „klasické“ geometrie značně lišily. Tyto poznatky, které byly původně velmi roztržité a zpočátku značně závislé na konkrétním hardware, se postupně dařilo systematizovat a pro takto nově vznikající obor se vžil název **počítačová grafika**.

S rozvojem možností hardware rostly i možnosti grafických systémů a jejich vývojáři byli stavěni před nové a nové problémy. Při vytváření stále složitějších objektů přestávaly dostačovat „elementární“ křivky, plochy a tělesa a „klasické“ geometrické operace. Křivky plochy a tělesa bylo třeba tvarovat uživatelsky intuitivní změnou polohy několika „významných“ bodů, plochy a tělesa bylo třeba tvořit předem definovaným pohybem křivek a ploch atd. Tyto úlohy jsou svojí povahou sice čistě geometrické, avšak bývají řešeny na přímou „objednávku“ počítačových grafiků. Jsou natolik specifické, že geometrie, která je řeší, se velmi často označuje jako **geometrie počítačová**.

Rozdíl mezi „klasickou“ **geometrií**, **počítačovou geometrií** a **počítačovou grafikou** je jasně vidět na následujícím jednoduchém příkladu. Počítačový grafik potřebuje algoritmus na vyplnění trojúhelníka barvou. Z geometrie ví, že spojíme-li vnitřní bod trojúhelníka s bodem vnějším, pak tato spojnice protne obvod trojúhelníka. Je to věc tak zřejmá, že se v klasické geometrii vůbec nedokazuje a stojí v jejích základech jako axiom. Algoritmus by tedy mohl fungovat tak, že by vzal vnitřní bod trojúhelníka (např. jeho těžiště) a postupoval od něj v libovolném směru k průsečíku s obvodem. Vzniklou úsečku by pak obarvil. Opakováním tohoto postupu pro jiné vnitřní body a jiné směry se pak docílí vybarvení celého trojúhelníka. Pokud by ovšem tato úloha byla skutečně takto naprogramována, skončila by v drtivé většině případů „vylitím“ barvy na celé výstupní okno. Proč? Jednoduše proto, že výstupní zařízení počítače není „klasická“ geometrická rovina a spousta takto sestrojených polopřímek obvod trojúhelníka neprotíná. Zde musí nastoupit počítačový geometr. Zahodit tento axiom a přijít na jiný geometrický princip, který by mohl nabídnout grafikovi k implementaci.

Grafické systémy jsou dnes většinou natolik intuitivní, že jejich základní ovládání lze pochopit i bez speciální geometrické přípravy. Takového uživatele moderního grafického systému však lze přirovnat ke svátečnímu řidiči, který usedne za volant formule 1. Při troše štěstí se mu možná podaří bez nehody pomaloučku projet prázdný závodní okruh (tj. v CADu možná vytvoří nějakou „docela hezkou věc“), ale závodit (sofistikovaně navrhnout třeba nový raketový motor) takový člověk prostě nemůže. Dobrý jezdec formule 1 musí být jednak výborným řidičem a jednak musí podrobně znát konstrukci svého vozu, aby mohl co nejlépe využít všech jeho možností. Dobrý konstruktér dnes musí být velmi dobrý geometr, aby měl předem jasnou představu o tom, co a jak chce vlastně tvořit, a detailně znát principy a algoritmy skryté v jeho grafickém systému, aby mohl využít všech jeho možností k realizaci svého záměru. Při studiu počítačové geometrie a počítačové grafiky Vám tedy přejeme hodně úspěchů.

## 1 Grafická data

Základní rozdíly mezi geometrií a počítačovou grafikou spočívají již v elementárních geometrických pojmech „bod“ a „přímka“. Podle **Euklida** (asi 325–265 př. n. l.), zakladatele systematické geometrie, je bod to, co nelze rozdělit na části (tedy „tečka bez rozměrů“) a přímka je délka bez šířky (tj. „nekonečně dlouhá, rovná a nekonečně tenká čára“). Zobrazovací plocha výstupního zařízení počítače (ať již monitoru či tiskárny, koneckonců i sítnice lidského oka) je však fyzické zařízení a „body bez rozměrů“ zobrazovat resp. vnímat neumí. Místo pojmu **bod** je proto používán pojem **pixel** (z anglického *picture element*) jako „nejmenší zobrazitelný útvar“. „Přímka“ na výstupním zařízení nemůže být ani nekonečná, ani rovná, ani nekonečně tenká. Informace, které tato zařízení poskytují, i informace, které jsou ukládány do počítače v podobě souborů, jsou konečné, přesněji řečeno jsou konečnou posloupností základních jednotek informace – **bitů**.

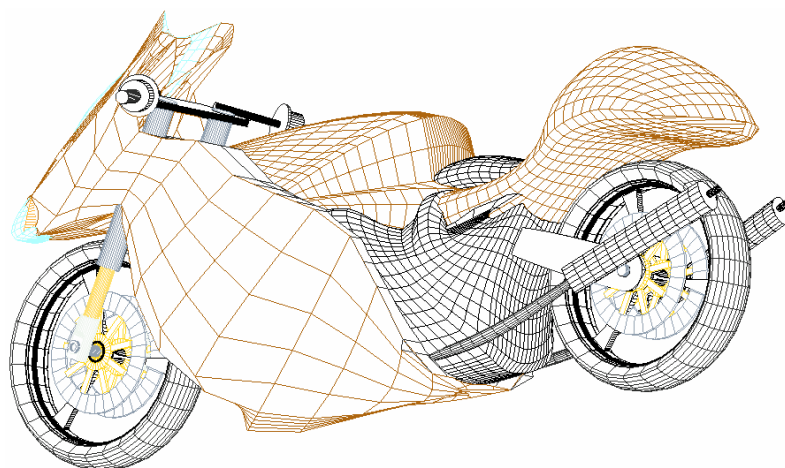
**Počítačová geometrie a grafika** se zabývá počítačovým zpracováním **grafických informací**, přičemž grafickou informací rozumíme reprezentaci lidského zrakového vjemu pomocí výše zmíněné konečné množiny bitů.

Základním pojmem počítačové geometrie i počítačové grafiky je pojem obraz. **Obrazem** rozumíme reprezentaci. Podle způsobu této reprezentace pak hovoříme o datech **vektorových** resp. **rastrových**.

### 1. 1 Vektorová a rastrová data

Grafické informace, data, která jsou uložena v souborech, se zásadně dělí na dvě velké skupiny - na data **vektorová** a data **rastrová**.

**Vektor** je v souvislosti s grafickými soubory chápán v „tradičním“ slova smyslu, tj. je ztotožňován s „orientovanou“ úsečkou, tj. úsečkou, na které rozlišujeme počáteční a koncový bod, popř. s veličinou která je určena velikostí, směrem a orientací. Pojem vektoru v počítačové grafice je poněkud šitší. V zásadě lze říci, že vektorový grafický soubor obsahuje informace o objektech složených z křivek a jednoduchých těles, které umožňují jejich geometrickou konstrukci. Je-li takto uložena např. kružnice, soubor neobsahuje informace o všech jednotlivých bodech, které na ní leží. Informuje o tom, že se jedná o kružnici, dále obsahuje souřadnice jejího středu, jednoho bodu, který na ní leží, a ve 3D modelářích je rovněž určena rovina její konstrukce. Připojeny mohou být rovněž informace

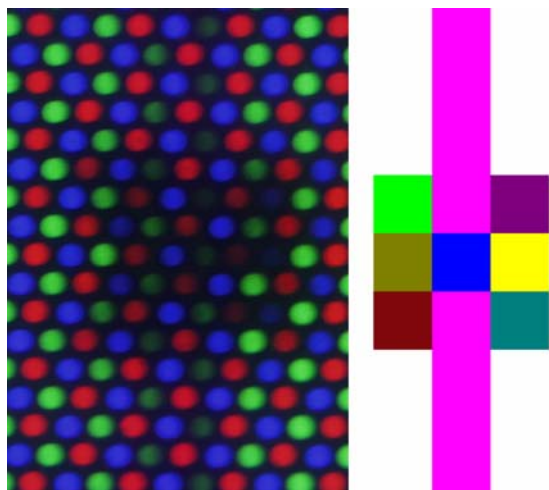
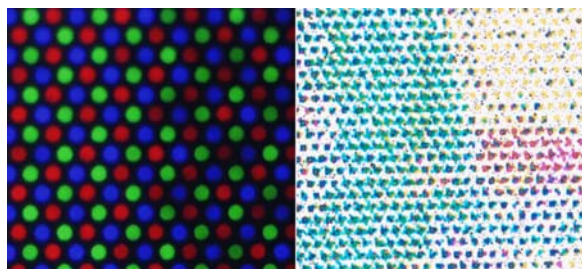


o barvě objektu a tloušťce čáry, kterou má být sestroyen. Program, pro který jsou tato data určena, musí být schopen tyto informace správně přečíst a musí obsahovat algoritmus, který na základě těchto informací kružnici sestrojí. Vektorová data jsou typická např. pro technické výkresy.



**Bitmapová (rastrová) data:** neobsahují „vektorové“ informace o uloženém objektu. Z těchto dat nelze (alespoň ne jednoduše) zjistit, zda se rastrový výstup - **obraz** skládá z obdélníků, kružnic, či jehlanů. Soubor obsahuje informace o velikosti obrazu, o způsobu případné komprese a kódování barev. Samotný obraz je uložen jako matice, jejíž každý prvek znamená jeden bod obrazu. Rastrově jsou ukládány buď informace, které již nebudou dále upravovány systémem,

kterým byly vytvořeny (např. žánrový pohled na strojní součást), nebo obrazy, které nebyly pořízeny počítačem (např. fotografie). Na rastrovém principu funguje většina zobrazovacích zařízení (monitory, jehličkové, inkoustové i laserové tiskárny, televize apod.).



Rastrová data ukládáme data jako souřadnice bodů, které se ovšem, jak již bylo řečeno, nezobrazují jako bezrozměrné objekty. Konkrétní podoba těchto útvarů (fyzických pixelů) závisí na výstupním zařízení. Na připojeném obrázku si můžeme prohlédnout fyzické pixely monitoru a inkoustové tiskárny. V počítačové geometrii jsou pixely modelovány jako čtverce, popř. obdélníky se společnými stranami. Takto modelované „fyzické pixely“ používají dnes prakticky všechny grafické aplikace.

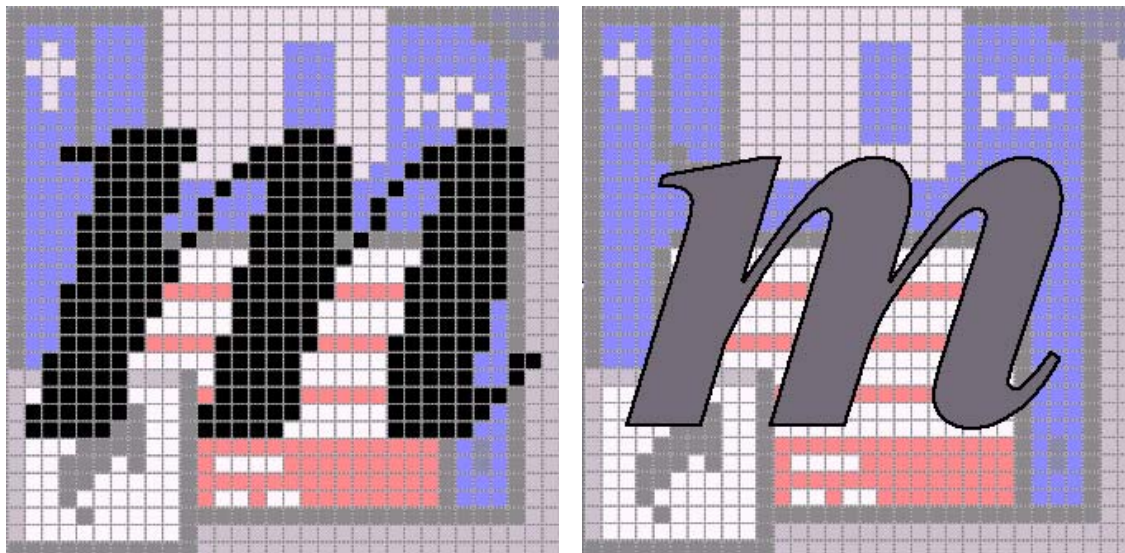
Důležitými vlastnostmi rastrového obrazu je **velikost pixelu** resp. **velikost obrazu**. Velikost jsme většinou zvyklí udávat v délkových jednotkách (např. v centimetrech). Nedocení významu velikosti fyzických pixelů může vést k zásadním omylům, a to i u profesionálních grafiků. Rastrová data jsou interpretována jako matice, jejichž každý prvek znamená jeden pixel obrazu. O kvalitě

výsledného výstupu tak mimo jiné rozhoduje velikost pixelů nebo častěji udávané rozlišení, tj. „počet bodů na palec“ - **DPI - Dots Per Inch** (počet bodů na palec).

Rozlišení 800×600 může poskytovat vysoce kvalitní (malý) obraz na čtrnáctipalcovém monitoru a díky omezené rozlišovací schopnosti lidského oka zde již nemá smysl použít rozlišení větší. Totéž rozlišení však bude nedostatečné na monitoru jednadvacetipalcovém. Obraz s rozlišením 2000×3000 bude kvalitně vytištěn na listu A4, na listu A3 bude kvalita téhož obrazu podstatně horší.

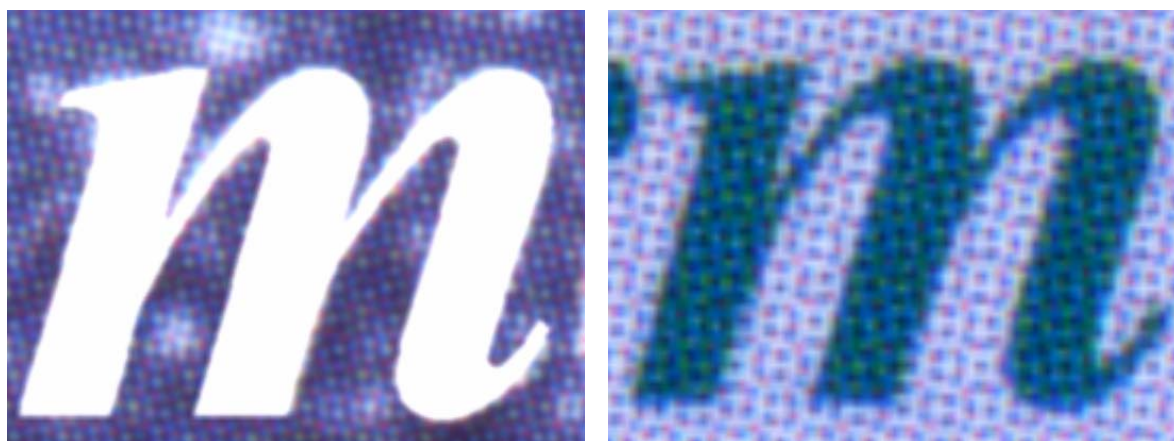


Vektorová a rastrová data se dnes velmi často ukládají do jednoho souboru pomocí tzv. vrstev. Vektorová data obsahují informace o objektech složených z křivek, ploch či jednoduchých těles, které umožňují jejich geometrickou konstrukci. „Fyzická velikost“ takto reprodukováného objektu (tj. velikost udávaná v délkových jednotkách) může být optimalizována dle konkrétního výstupního zařízení. Tuto situaci ilustruje připojený obrázek.



Je-li do rastrových dat s nízkým rozlišením (např. do obrázku) dodáno písmo a vše uloženo jako rastr, pak písmo nemůže mít větší rozlišení, než původní obrázek. Je-li však obrázek uložen jako rastr a písmo ve vektorové podobě, může být text dle možností výstupního zařízení vytištěn podstatně kvalitněji, než obrázek, do kterého je vložen. Odtud pramení názor, že pro tiskařské účely jsou vektorová data vhodnější. Většinou slyšíme, že pro tiskařské účely je třeba text dodávat vždy vektorově, neboť rastrová písmena jsou „zubatá“, „chlupatá“ apod. Tento názor zastává řada tiskařů, počítačových grafiků a průmyslových designérů. Tento názor by však byl správný pouze v případě, že lidské oko má nekonečnou rozlišovací schopnost. Nekonečné by muselo být rovněž rozlišení tiskárny a osvitového zařízení a čtyři fólie s barevnými složkami by se musely být položeny s přesností nejméně na průměr atomového jádra. Žádná z těchto podmínek však ani zdaleka nemůže být splněna.

Na těchto obrázcích můžeme porovnat výsledky reálného tiskařského procesu. Jedno písmeno je rastrové, bylo uloženo spolu s obrazem s rozlišením 600 DPI. Druhé je do obrázku se stejným rozlišením dodáno vektorově. Při této kvalitě vstupních dat nejsme schopni zaznamenat rozdíl v písmu, (natož „zubatost“ či „chlupatost“) ani na mikrofotografii pořízené s rozlišením 3600 DPI.



## 1. 2 Barevné systémy

Dalším podstatným rozdílem mezi geometrickou a grafickou informací je barva. Klasické geometrické konstrukce definují vzájemnou polohu event. velikost objektů, které mají být sestrojeny, nikdy však jejich barvu. Barva je ale důležitou zrakovou informací a problematika její reprezentace tvoří nedílnou součást počítačové geometrie a grafiky.

Lidské oko je omezeno ve své činnosti nejen co do velikosti vnímaných objektů, ale také co do počtu barev. Barevný vjem vzniká podráždění specializovaných buněk lidského oka (čípků) elektromagnetickým vlněním v rozsahu vlnových délek 380 – 720 nm. Podle vlnové délky světelného zdroje rozlišujeme světlo **monochromatické** (obsahuje v ideálním případě jedinou vlnovou délku) a **achromatické** (obsahuje v ideálním případě všechny vlnové délky viditelného světla). Vnímané světlo je charakterizováno několika základními vlastnostmi:

**a) Barva (hue odstín)** – je dána vlnovou délkou s největší intenzitou

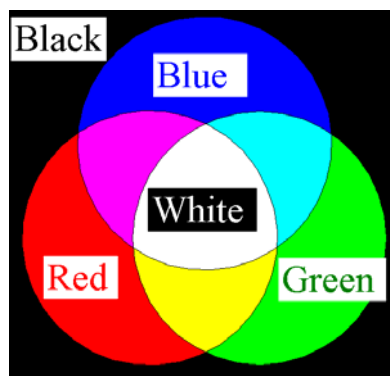
**b) Sytost (saturation - saturace)** – závisí na intenzitě achromatické složky světla (tj. vlnovými délkami mimo maximum)

**c) Jas (value - světelnost)** – je dán celkovou intenzitou světla

Každý pixel v bitmapovém souboru je reprezentován určitým počtem bitů. Tím je určen počet barev, které může daný pixel nabýt. Je-li  $n$  počet bitů na pixel, je možno zobrazit  $2^n$  barev. Z historie známe dvouúrovňová (jednobitová) zařízení, která dovozovala zobrazovat dvě úrovně (monochromatické monitory, jehličkové tiskárny). Lidské oko má konečný počet barevných receptorů, které obsáhnou rozsah světelných frekvencí 380-770 nm. Tvrdí se, že lidské oko dovede rozeznat až deset milionů barev (přesný počet se v individuálních případech samozřejmě liší). Zařízení, které je schopno zobrazit  $2^{24} = 16\,777\,216$  barev, považujeme proto za **true color** (pravé barvy). True color zařízení tedy potřebuje  $24 = 3 \cdot 8$  bitů = 3byty na pixel. V minulosti byla termínem true color označována i zařízení o 15 resp. 16 bitech na pixel (32768 resp. 65536 barev).

Velikost fyzického pixelu na obrazovce je závislá na velikosti monitoru a pohybuje se řádově v desetinách milimetru. Pokud je vedle sebe umístěno několik takových pixelů o různých barvách, pak za normálních podmínek není oko schopno rozlišit jeden pixel od druhého. Zraková informace je v mozku integrována a my vnímáme třeba i barvu, která na výstupním zařízení vůbec není zobrazena. Důležitou věcí, která hraje roli při vnímání barev, je způsob, jakým je barva vytvořena. Je všeobecně známo, že barvy lze „míchat“. Smíchání červené a zelené vzniká žlutá, modrá a žlutá dává zelenou atd. V barevných modelech jsou proto barvy realizovány mícháním základních barev.

Rozlišujeme dvě základní skupiny barevných systémů: **aditivní a subtraktivní**. V aditivních systémech je „nepopsaný“ podklad černý a barvy vznikají přidáváním základních barev.

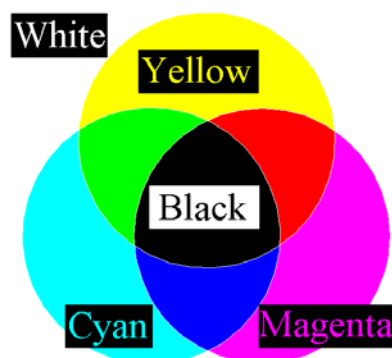


Přítomnost všech základních barev v plné intenzitě dá (teoreticky) barvu bílou. Příkladem mohou být barvy na monitoru. V subtraktivním systému je podklad bílý a barvy vznikají odečítáním od bílé. Přítomnost všech základních barev v plné intenzitě dá (teoreticky) barvu černou.

**RGB (Red-Green-Blue):** je aditivní tříbarevný systém, každý pixel je reprezentován trojicí barev - červenou, zelenou a modrou (viz první obrázek). Pro 24-bitovou reprezentaci znamená (0,0,0) černou, (255,255,255) bílou. Barevný prostor RGB, kde každá barevná složka má k dispozici jeden byte, si

Ize představit jako krychli o straně 256 (tzv. RGB krychli), jejíž vrcholy tvoří barvy černá, bílá, červená, zelená, modrá, modrozelená, fialová a žlutá.

Velikost paměti současných grafických karet většinou umožňuje reprezentovat pixel čtyřmi byty – můžeme se setkat se zkratkou **RGBA**, kde RGB značí výše popsany barvný model a čtvrtý byte je tzv. alfa kanál, který obsahuje informaci o průhlednosti daného pixelu. Používá se při skládání více obrazů do jednoho celku. RGBA tedy není nový barevný model, ale znamená přidání další informace o jednotlivých pixelech. Model RGB je v současné době standardizován. Standardizace modelu RGB (model **sRGB**) spočívá v přesné definici základních barev Red, Green, Blue a rovněž v přesné definici černé a bílé. Na každém RGB zařízení, kde těchto šest barev splňuje standard sRGB, se pak barva daná vektorem  $\mathbf{r} = (r; g; b)$  zobrazuje stejně. Podporuje ho stále více monitorů, digitálních fotoaparátů a



dalších zařízení, proto je v současné době velmi rozšířen.

**CMY** (Cyan - Magenta - Yellow) Tento systém je subtraktivní, barvy se zobrazují inkoustem nebo emulsi na bílém povrchu. Je používán na většině inkoustových a laserových tiskáren. Jestliže emulsi pohlcuje červenou, jeví se v bílém světle jako azurová, neboť odráží zelenou a modrou. Pohlcení zelené vidíme jako purpurovou, je-li pohlcena modrá, vidíme žlutou. Jsou-li emulsi pohlcovány všechny složky, vzniká (teoreticky) černá.

Je samozřejmě velmi žádoucí, aby všechny barevné systémy byly schopny poskytnout všechny barvy, které je lidské oko schopno vnímat. Systémy, které jsou schopny vzájemného převodu při poskytování stejných barev, nazýváme **reverzibilní**. V reverzibilních barevných systémech existuje vyjádření téže barvy téhož jasu a téže saturace. Systémy RGB a CMY jsou reverzibilní, přepočítání mezi nimi je velmi jednoduché. Je-li  $\mathbf{r} = (r; g; b)$  vektor vyjadřující barvu v prostoru RGB, pak stejná barva v prostoru CMY je dána vektorem  $\mathbf{c} = (c; m; y)$ , pro který platí (pracujeme v true color systému, tj. 255 hodnot na jednu barevnou složku)

$$\mathbf{c}^T = \mathbf{M} \cdot \mathbf{r}^T + \mathbf{p} \quad (1)$$

$$\begin{pmatrix} c \\ m \\ y \end{pmatrix}^T = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} r \\ g \\ b \end{pmatrix}^T + \begin{pmatrix} 255 \\ 255 \\ 255 \end{pmatrix} = \begin{pmatrix} -r \\ -g \\ -b \end{pmatrix}^T + \begin{pmatrix} 255 \\ 255 \\ 255 \end{pmatrix}$$

Zpětný přepočítání barvy z CMY do RGB obdržíme tak, že z rovnice (1) vyjádříme vektor  $\mathbf{r}^T$ :

$$\begin{aligned} \mathbf{M} \cdot \mathbf{r}^T + \mathbf{p} &= \mathbf{c}^T \\ \mathbf{M} \cdot \mathbf{r}^T &= \mathbf{c}^T - \mathbf{p} \quad \text{zleva} \\ \mathbf{M}^{-1} \cdot \mathbf{M} \cdot \mathbf{r}^T &= \mathbf{M}^{-1} \cdot (\mathbf{c}^T - \mathbf{p}) \\ \mathbf{r}^T &= \mathbf{M}^{-1} \cdot (\mathbf{c}^T - \mathbf{p}) \end{aligned}$$

Snadno se přesvědčíme o tom, že v našem případě je  $\mathbf{M} = \mathbf{M}^{-1}$ , takže pro převod z CMY do RGB je

$$\begin{pmatrix} r \\ g \\ b \end{pmatrix}^T = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \left[ \begin{pmatrix} c \\ m \\ y \end{pmatrix}^T + \begin{pmatrix} 255 \\ 255 \\ 255 \end{pmatrix} \right] = \begin{pmatrix} 255 - c \\ 255 - m \\ 255 - y \end{pmatrix}$$

**CMYK** (Cyan - Magenta – Yellow - Black): Při tisku jsou barevné obrazy vytvářeny jako soubor několika obrazů tvořených několika základními barvami na průhledné fólii. Základní barevné pigmenty však nemohou být dokonale krycí, protože by barvy nebylo možné prošlým světlem míchat. Základní barvy tak obsahují jistý podíl bílého světla a z důvodů nedostatečné saturace není možné jejich mícháním obdržet dokonale černou barvu. Z tohoto důvodu se tedy černá tiskne jako samostatná barva. Tu lze použít i ke ztmavení základních barev.

Model CMYK je tvořen čtyřmi základními barvami, vektor  $\mathbf{c}_k = (c; m; y; k)$  je čtyřsložkový, matice  $\mathbf{M}_k$  není tedy čtvercová a prostory RGB a CMYK nejsou ekvivalentní. Díky menší saturaci barev prostoru CMYK oproti RGB nelze v CMYK reprodukovat přesně barvy s velkou saturací, tedy především vrcholy RGB krychle (kromě černé a bílé). Množina barev, které je daný barevný systém schopen přesně reprodukovat, tvoří tzv. **gamut** barevného systému. Ani gamut RGB neobsahuje všechny barvy, které je lidské oko schopno vnímat. Gamut systému CMYK je ještě podstatně užší a do značné míry závisí na možnostech použitého osvitového zařízení. Pro převod RGB  $\rightarrow$  CMYK se proto používají nejrozličnější matice  $\mathbf{M}_k$  (tzv. profily CMYK), které se snaží nutně rozdíly barev nejrozličnějším způsobem minimalizovat. Nevhodný profil může při převodu obraz zcela znehodnotit. To se bohužel velmi často děje a bývá to zdrojem nedorozumění mezi poskytovatelem primárních dat a grafikem typografické dílny.

**Systém YUV:** Systém sRGB poskytuje velmi širokou paletu barev, je však poměrně nevýhodný pro bezztrátovou kompresi (sousední pixely mají velmi často velmi blízké hodnoty). To je velká nevýhoda např. při přenosu televizního signálu. Model YUV tuto nevýhodu do značné míry potlačuje. Odděluje jasovou složku Y - jas a dvě barevné složky U a V (písmeno Y tedy nezaměňovat s Y v modelu CMY, kde značí žlutou). Vektor  $\mathbf{y} = (y; u; v)$  obdržíme z vektoru  $\mathbf{r} = (r; g; b)$  opět pomocí rovnice tvaru (1), kde  $\mathbf{p} = (0; 0; 0)$ , tedy

$$\mathbf{y}^T = \mathbf{M} \cdot \mathbf{r}^T : \begin{pmatrix} y \\ u \\ v \end{pmatrix}^T = \begin{pmatrix} 0,299 & 0,587 & 0,144 \\ -0,147 & -0,289 & 0,436 \\ 0,615 & -0,515 & -0,100 \end{pmatrix} \begin{pmatrix} r \\ g \\ b \end{pmatrix}^T \quad (3)$$

Matice převodu je regulární, existuje k ní tedy matice inverzní, tudíž rovněž jednoznačný převod zpět z YUV do RGB. Nalezneme ho řešením maticové rovnice (3):

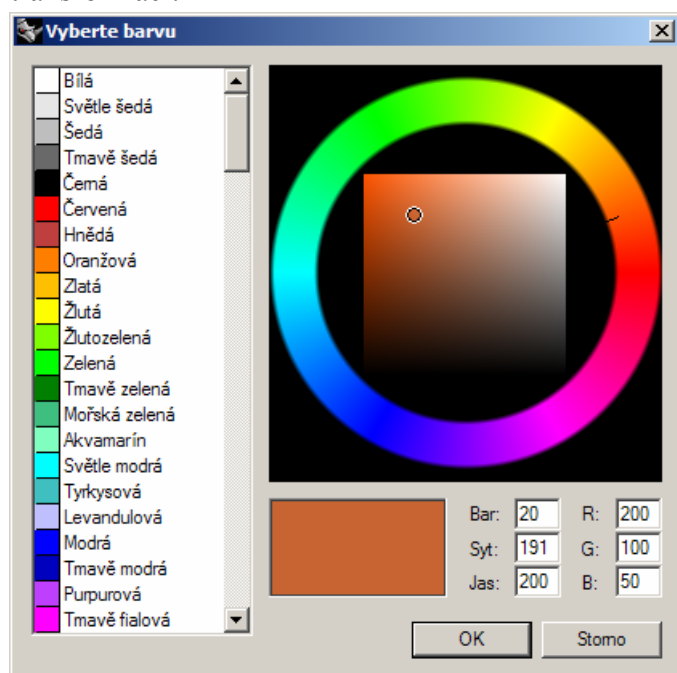
$$\left. \begin{array}{l} \mathbf{M} \cdot \mathbf{r}^T = \mathbf{y}^T / \cdot \mathbf{M}^{-1} \text{ zleva} \\ \mathbf{M}^{-1} \cdot \mathbf{M} \cdot \mathbf{r}^T = \mathbf{M}^{-1} \cdot \mathbf{y}^T \\ \mathbf{r}^T = \mathbf{M}^{-1} \cdot \mathbf{y}^T \end{array} \right\} \Rightarrow \begin{pmatrix} r \\ g \\ b \end{pmatrix}^T = \begin{pmatrix} 0,299 & 0,587 & 0,144 \\ -0,147 & -0,289 & 0,436 \\ 0,615 & -0,515 & -0,100 \end{pmatrix}^{-1} \cdot \begin{pmatrix} y \\ u \\ v \end{pmatrix}^T = \begin{pmatrix} 1,000 & 0,000 & 1,137 \\ 1,000 & -0,397 & -0,580 \\ 1,000 & 2,034 & 0,000 \end{pmatrix} \cdot \begin{pmatrix} y \\ u \\ v \end{pmatrix}^T$$

(vypočítejte inverzní matici!)

**Systémy HSV; HSL** – jsou ekvivalentní s RGB. **HSV** (Hue-Saturation-Value) transformuje RGB krychli na pravidelný šestiboký jehlan, v němž bílá tvoří střed podstavy, černá vrchol jehlanu a ostatní vrcholy RGB krychle se zobrazí na vrcholy podstavy jehlanu. Je to barevný systém, jehož složky respektují vymezení barvy tak, jak bylo uvedeno na začátku této



kapitoly. Jas klesá se vzdáleností od podstavy, ve vrcholu jehlanu je nulový. Saturace a barevný tón tvoří jakousi obdobu polární soustavy souřadné, kde ovšem barvy se stejnou saturací neleží na kružnici, ale na obvodu pravidelného šestiúhelníka. Tuto nepříjemnost odstraňuje model **HSL** (Hue-Saturation-Lightness). Barevný prostor má tvar dvojkužele, kde saturace a barevný tón tvoří (tentokrát klasickou) polární soustavu souřadnou a barvy v rovině podstavy mají poloviční světelnost. Ta se vzdáleností od podstavy roste směrem k „bílému“ vrcholu a klesá směrem k „černému“ vrcholu. V grafických systémech využívajících systémy HSV resp. HSL nenajdeme většinou jehlan či dvojkužel, ale některou z jejich možných transformací.



Na připojeném obrázku vidíme implementaci systému **HSL** v grafickém systému **Rhinoceros**. Barevný odstín ( $H$ ) vybíráme na kružnici po jednom stupni – může tak nabýt tři sta šedesáti hodnot. Sytost ( $S$ ) zvyšujeme na vodorovné straně čtverce zprava doleva a jas ( $L$ ) zdola nahoru.

Přepočty RGB na HSL a naopak jsou již komplikovanější. Odstíny šedé, tj. případy kdy  $R = G = B$  se přepočítávají na  $H = S = 0$  a  $L = R = G = B$ . V ostatních případech se postupuje následovně: V HSL se položí

$$L = \max \{R; G; B\}.$$

$$S = \frac{L - \min \{R; G; B\}}{L} \cdot 255$$

Konečně výpočet úhlu pro barvu ( $H$ ):

$$L = R \quad \Rightarrow \quad H = 60 \cdot \frac{G - B}{L - \min \{R; G; B\}}; \quad H < 0 \Rightarrow H := 360 + H$$

$$L = G \quad \Rightarrow \quad H = 120 \cdot \frac{R - B}{L - \min \{R; G; B\}}$$

$$L = B \quad \Rightarrow \quad H = 270 \cdot \frac{R - G}{L - \min \{R; G; B\}}$$

**1. Příklad:** převedme barvu  $(R; G; B) = (200; 100; 50)$  na  $(H; S; L)$ .

**Řešení:**  $L = \max \{R; G; B\} = R = 200$

$$S = \frac{L - \min \{R; G; B\}}{L} \cdot 255 = \frac{200 - 50}{200} \cdot 255 \approx 191$$

$$H = 60 \cdot \frac{G - B}{L - \min \{R; G; B\}} = 60 \cdot \frac{100 - 50}{200 - 50} = 20$$